

AD-A260 499



S **ELECTE** **D**
FEB 9 1993
c

Technical Document 2448
January 1993

Evaluations Using the Software Engineering Institute (SEI) Environments Criteria

S. A. Parker

93-02331
93-02331



5307

Authorized for public release; distribution is unlimited.



Technical Document 2448
January 1993

**Evaluations Using the
Software Engineering Institute (SEI)
Environments Criteria**

S. A. Parker

**NAVAL COMMAND, CONTROL AND
OCEAN SURVEILLANCE CENTER
RDT&E DIVISION
San Diego, California 92152-5000**

J. D. FONTANA, CAPT, USN
Commanding Officer

R. T. SHEARER
Executive Director

ADMINISTRATIVE INFORMATION

The work described in this report was done under the Software Engineering Environment Prototypes (SWEEP) task of the software engineering for the Command, Control, and Communications (C3) Systems project. The SWEEP task is part of the Computer Technology block program sponsored by the Office of Naval Technology. The work was done during FY 1992 by S. A. Parker of the Computer Software and Technology Branch, Code 411, at the Naval Command, Control and Ocean Surveillance Center, RDT&E Division. Funds for this work were provided through program element 0602234N.

Released by
G. Schulte, Head
Computer Software and Technology
Branch

Under authority of
A. G. Justice, Head
Information Processing
and Displaying Division

ACKNOWLEDGMENTS

The author thanks Phillipe Collard of Telesoft, Sandra Swimmer of the Rational Corporation, and Andy Taylor of Verdix for their technical assistance and for reviewing this document. The author also thanks Michael Shapiro and Sharon Rotter of NRaD for reviewing this document.

CONTENTS

INTRODUCTION	1
APPROACH	1
CRITERIA RECOMMENDATIONS	3
LESSONS LEARNED	3
CONCLUSIONS	4
REFERENCE	5
BIBLIOGRAPHY	5
APPENDICES	
A SEI CRITERIA	A-1
B ENHANCEMENTS TO THE SEI CRITERIA	B-1
C RISCAda EVALUATION	C-1
D RATIONAL EVALUATION	D-1
E SUN Ada DEVELOPMENT ENVIRONMENT EVALUATION	E-1
F REFERENCES	F-1

DTIC QUALITY INSPECTED 3

Accession For	
NTIS	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

INTRODUCTION

This report describes the investigation done under the Software Engineering Environment Prototypes (SWEEP) task of the software engineering for the Command, Control, and Communications (C3) Systems project. The SWEEP task is part of the Computer Technology block program sponsored by the Office of Naval Technology.

The goal of this investigation was to find weaknesses in the Software Engineering Institute's (SEI) criteria for evaluating the software engineering environments (SEEs) detailed by Smith (1991). The SWEEP task also sponsored the SEI's development of the criteria. During this investigation, the criteria were used to evaluate three commercial software engineering environments. These evaluations found a few weaknesses in the SEI criteria and provided insights into the environments being investigated.

This research was carried out by a software engineer from the Computer Software and Technology Branch (Code 411) at the Naval Command, Control and Ocean Surveillance Center, RDT&E Division (NRaD), who had substantial assistance from commercial vendors. The Rational Corporation vendor provided support with on-site classroom training on the Rational environment. All vendors provided on-call consultation to answer questions about their environments.

This report includes the following:

- A description of the approach used for the evaluations.
- Lessons learned about the criteria and environments.
- Recommendations for changes to the SEI criteria.
- The SEI environment evaluation criteria.
- Three environment evaluations.

Information in this report was provided to the SEI to help modify its evaluation criteria. Other potential users of this report are project personnel who either use or plan to use any of the evaluated environments.

APPROACH

The objective of this task was to investigate the SEI's environment evaluation criteria to determine their strengths and weaknesses. The results of this investigation will be used to improve the criteria before the SEI's final report is issued in 1993. This evaluation was conducted with the criteria in the SEI draft dated December 1991. A side effect of the investigation was the evaluation of three commercial software environments.

The SEI evaluation criteria consisted of three sections: the first two sections covered an organization's readiness to use a new environment and the third section contained the

criteria for evaluating SEEs. The evaluations were performed by using a few questions from sections 1 and 2 and all the questions from section 3 of the SEI draft report. The criteria from all SEI draft report sections are in appendix A.

NRaD reviewed the SEI evaluation criteria to determine whether obvious weaknesses or areas of concern existed. Then the engineer assigned to the task, or task engineer (TE), determined the environments to evaluate. Of the many commercial SEEs at NRaD, three were chosen to be evaluated because they were already on Code 411 computers. The three environments were the Rational, Telesoft's RISCAda, and the Sun Ada Development Environment.

The first step in evaluating RISCAda and the Sun Ada Development Environment was to read user's and reference manuals to determine preliminary responses to criteria questions. Answers were confirmed by using the environments. Source code for a Code 411 project, the Joint Automated Message Editing System (JAMES), which had been developed for the Defense Information Systems Agency, was loaded into each environment. The JAMES consisted of approximately 18,000 lines of Ada code. Loading it into both environments and using the environments' tools to manipulate the code helped answer questions about the criteria.

The Rational environment was evaluated in a slightly different manner because training was provided when the environment was purchased. The TE was able to answer most criteria questions from the information provided during training. Answers were confirmed by using the environment during the hands-on portion of training.

After he performed each environment evaluation, the TE still had unanswered questions. The unanswered questions usually concerned the internal workings of the environment. The vendors helped answer these questions.

When the evaluations were complete, they were reviewed by the vendors. Vendor reviews were conducted for two reasons. First, vendors could discuss how well the criteria differentiated their product from other environments on the market. Second, vendors could verify the evaluations' results. These evaluations are in appendices C, D, and E. These evaluations will be submitted for inclusion in the environment data base being developed by the Software Technology Support Center at Hill Air Force Base, Utah.

The vendors' review of the evaluations was helpful. The vendors found a few errors and corrected some misconceptions. However, their corrections were not always included in the evaluations contained in this report. For example, one vendor said that several criteria questions were not applicable to his environment because the environment was not intended to provide the capabilities discussed. The original responses, which reflected that the environment did not provide the capabilities in question, were retained in the evaluations. Users of the evaluations would be interested in whether the environment provided these capabilities, and this particular environment did not provide these capabilities.

Throughout the investigations, the TE collected environment evaluation criteria from other sources, including International Software Systems, Inc., and the Air Force's Software Technology Support Center at Hill Air Force Base. These other criteria were compared to the SEI criteria to find additional weaknesses and provide the SEI with feedback for possible improvement. At the same time, the authors of the other criteria were given the SEI criteria to review and comment on. The Air Force's Software Technology Support Center provided useful comments on the criteria and gave advice on performing the evaluations.

CRITERIA RECOMMENDATIONS

The criteria developed by the SEI were well conceived and comprehensive. The explanations provided for each question were especially helpful; they clarified the question and suggested the range of possible answers the evaluator could give. Most of the environment capabilities and operations were well covered. No major deficiencies were identified. Some areas of the criteria did need improvement before publication of the final draft. These improvements included minor changes in wording and the addition of some questions. NRaD's recommendations, along with the justifications for them, were given to the SEI prior to the publication of this report. The changes will be incorporated into the SEI's final draft of the criteria, which will be published at about the same time as this report.

Based on the evaluations' results, the major suggestions given to the SEI were the following:

- Add a checklist of functional capabilities to the criteria report. This checklist will let a manager who is reviewing evaluations quickly select the environments with the required characteristics.
- Add more questions on the user interface.
- Add questions on documentation and configuration control reports.

The detailed recommendations are in appendix E.

LESSONS LEARNED

During this investigation, the TE learned several things that may help future environment evaluators.

1. "Environment" means different things to different people. To some tool vendors, it is a compiler (and its associated tools) with an X Window interface. To others, it is an organized and integrated group of tools that support the entire software life cycle. Thus, the evaluator should expect the range of capabilities offered by different environments to vary substantially.

2. "Tool integration" also has many possible meanings. The vendor's definition can vary significantly from the evaluator's definition. Evaluators should determine their definition of "tool integration" and use it to rate the environment's tool integration capability.
3. If the vendor offers a training course on the environment being evaluated, the evaluator should take it. Taking a course leads to an easy way to conduct evaluations. The evaluator should be aware, however, that in some cases the cost of the training class may be prohibitive. This is particularly true when the course evaluates a number of environments.
4. While most evaluation questions can be answered by reading the environment's documentation, the answers should be verified by using the environment. The documentation is occasionally misleading. The evaluator's interpretation of the documentation is occasionally wrong.
5. Vendor assistance will be required to answer some questions. The evaluator may choose to rate the vendors on how well and how readily they provide this assistance.
6. Vendors are usually glad to answer questions. They are also willing to look over evaluations and verify the results obtained.
7. Vendors are trying to sell their products. Evaluators are trying to evaluate the products fairly. The two goals may be in conflict. Evaluators should keep this in mind.
8. During the evaluation process, always use a software application that is representative of the applications that will be developed using the environment to test the environment. Evaluators may not get valid results if their typical applications are 100,000 lines of code while the evaluation uses 1000 lines of code.
9. Some significant deficiencies may be found in the environments during the evaluations. Report these problems to the vendors. The vendors appreciate the feedback and the opportunity to improve their products.
10. The evaluations in appendices C through E required 3 to 5 weeks per environment. Less time is required when a training class is taken.

CONCLUSIONS

The SEE evaluation criteria report developed by the SEI for this task is comprehensive. It appears to cover the range of capabilities offered by commercial environments. The number of questions is appropriate for conducting high-level evaluations in a timely manner. The criteria report contains a helpful explanation for each question, and it covers

the range of features and services required in a software engineering environment. The criteria has a few weak points but no major flaws. Suggestions for improving the criteria are in the Criteria Recommendations section and appendix B.

REFERENCE

Smith, D., C. Huff, E. Morris, and P. Zarrella. 1991. "Evaluation of Software Engineering Environments," Draft (December). Software Engineering Institute, Pittsburgh, PA.

BIBLIOGRAPHY

Barney, D., J. D'Asto, and R. Peterson. 1992. "Software System Development Test and Evaluation Plan and Test Results Report," (June). Software Technology Support Center, Hill Air Force Base, UT.

Harrison, T. "Evaluation Criteria for COTS Software Engineering Frameworks," Draft. International Software Systems, Inc., Austin, TX.

Rational Corporation. 1991. "Rational Environment Training: Basic Product Training Series Fundamentals," (April). Rational Corporation, Santa Clara, CA.

Rational Corporation. 1991. "Rational Environment Training: Subsystems and Configuration Management," (June). Rational Corporation, Santa Clara, CA.

Software Technology Support Center. 1992. "Software Engineering Environment Evaluation Criteria," Draft (April). Software Technology Support Center, Hill Air Force Base, UT.

Sun Microsystems. 1991. "Sun ADA User's Guide," 800-5584-10 (April). Sun Microsystems, Mountain View, CA.

Sun Microsystems. 1990. "Sun APSE User's Guide," 800-4882-05 (October). Sun Microsystems, Mountain View, CA.

Telesoft. 1991. "ARCS Reference Guide," 1852N-V1.2 (August). Telesoft, San Diego, CA.

Telesoft. 1991. "ARCS User's Guide," UG-1851N-V1.2 (August). Telesoft, San Diego, CA.

APPENDIX A. SEI CRITERIA

The criteria listed in this appendix come from the SEI environment evaluation criteria report (Smith, 1991). The SEI report also includes an explanatory paragraph for each question.

1.0 ORGANIZATIONAL READINESS

1.1 Management Commitment

- 1.1.1 From what level of management is the commitment to the SEE effort?
- 1.1.2 Does management have a personal stake in the success or failure of the new SEE?
- 1.1.3 What costs have been considered?
- 1.1.4 Is the level of this cost a "showstopper"?
- 1.1.5 Is management willing to fully fund the effort to cover the anticipated costs? If not, why not?
- 1.1.6 Has management planned a reserve of funds to cover some reasonable level of unanticipated costs?
- 1.1.7 How long is management willing to wait before seeing concrete results?
- 1.1.8 What are management's expectations on its return on investment?

1.2 Personnel Readiness

- 1.2.1 What is the range of experienced project personnel on the current development SEE?
- 1.2.2 How relevant will this experience be in the "new SEE"?
- 1.2.3 What training or experience do project personnel have on new methods that may be required by the "new SEE"?
- 1.2.4 What is the expected amount of training required for the "new SEE" for current personnel and new hires?
- 1.2.5 How receptive are the personnel to making changes that affect their working environment?
- 1.2.6 What is the perception of personnel to past changes in their working environment?
- 1.2.7 Does the organization "understand" the reasoning behind the change?

1.3 Base Computing Environment

1.3.1 Will the anticipated SEE require a significant upgrade or change in the current development environment?

1.3.2 How long will it take to install and test the new hardware and software plus train users?

1.3.3 How long will it take the average user to become proficient in the new base computing environment?

1.3.4 If there is a support organization for the base computing environment, has the impact of the changes been considered from its point of view?

1.3.5 Is the proposed computing environment "stable," or will it be subject to change?

1.4 Software Process

1.4.1 Do the planned changes simply automate the current development process? How transparent are these changes to the users?

1.4.2 If not, how are the current processes impacted?

1.4.3 How flexible and adaptable will the software process be in the new SEE?

1.4.4 Does the environment provide process-tuning capabilities?

2.0 OPERATIONAL ISSUES

2.1 Product Characteristics

2.1.1 What is the software product domain?

2.1.2 Are there contractual requirements for a specific development language?

2.1.3 What is the life expectancy of the software product?

2.1.4 What is the size of the code to be developed or managed?

2.1.5 Are there specialized host-target requirements?

2.2 Project Characteristics

2.2.1 Is there a need for a spectrum of development teams?

2.2.2 Is there a need for classified development?

2.2.3 What is the life expectancy of the software project?

2.2.4 Are there any specific project-level performance requirements?

2.3 Integration Requirements

2.3.1 What degree of integration is required/expected for data, control, presentation, and process?

2.3.2 What degree of coupling is there between the data, control, presentation, and process integration requirements and mechanisms?

2.3.3 Can commercial off-the-shelf tools be integrated directly into the environment?

2.3.4 What degree of integration is required/expected between development tools and methods?

2.3.5 What are the technology limitations of the integration requirements?

2.4 Vendor Support

2.4.1 What type of support is expected/needed from the environment vendor?

2.4.2 What is the stability of the vendor?

2.4.3 What is the stability of the environment?

2.4.4 What level of integration support is expected/needed from the vendor?

2.4.5 What is the level of product documentation quality?

3.0 ENVIRONMENT TECHNOLOGY

3.1 Data Repository Services

3.1.1 Is the repository based on a common data storage mechanism?

3.1.2 Is the repository isolated from the tools by the environment?

3.1.3 Are tool changes necessary to support central data storage?

3.1.4 Does the repository provide consistency checking and update services?

3.1.5 How are changes propagated?

3.1.6 Can the repository be distributed? Is there a data location service?

3.1.7 Is all data "live," or is there an archival service?

3.1.8 Can new data types be defined? New attributes?

3.2 Data Integration Services

3.2.1 Does the repository utilize a common data model?

3.2.2 Can the organization query the data model? Update?

3.2.3 Is there a registration/monitoring service for data update? Is it configurable?

3.2.4 How are complex data queries handled?

3.2.5 Are configuration services integrated with versioning services?

3.2.6 Can subenvironments be defined? Distributed? Configured?

3.2.7 Can data be exchanged with "foreign" environments?

3.3 Tools

3.3.1 Which functions are under tool control, and which are under environment control?

3.3.2 What expectations are made of tools?

3.3.3 Can tools be integrated at varying levels?

3.3.4 What tools are currently integrated into the environment?

3.3.5 Who maintains tool integrations?

3.3.6 What techniques are used to ensure tool replaceability?

3.4 Task Management Services

3.4.1 Does the environment provide services for task control?

3.4.2 How flexible are the control mechanisms?

3.4.3 How simple is the establishment of the process model?

3.4.4 Can a subenvironment be specified for individual tools?

3.4.5 Can roles be defined for individual users?

3.5 Message Services

3.5.1 What type of message service is provided?

3.5.2 Are message services handled as "transactions"? Can messages be canceled? Re-sent?

3.5.3 Can a "process flow" be defined?

3.5.4 Can message services span environment instances?

3.5.5 Can third-party tools be integrated into the system?

3.6 User Interface Services

3.6.1 Is the windowing system in keeping with industry trends?

3.6.2 What services are offered?

3.6.3 How are external tools integrated into the interface?

3.6.4 How flexible is the user interface?

3.7 Security

3.7.1 What types of security are provided?

3.7.2 Are reports provided?

3.7.3 At what level of granularity is security provided?

3.8 Framework Administration and Configuration

3.8.1 What personnel are required for administration?

3.8.2 Is training provided?

3.8.3 How easily can the environment be installed? Rebuilt? Upgraded?

3.8.4 Is the environment stable?

3.8.5 What enhancements are envisioned?

3.8.6 Are utilities provided for conversion from other tools and environments?

3.9 Process Encoding

3.9.1 What mechanisms are used for process encoding?

3.9.2 Is the environment oriented toward any one process? Portion of the life cycle?

3.9.3 How flexible is process support?

3.9.4 What level of effort is required to encode an organization's process?

3.9.5 How complete is the process encoding?

APPENDIX B. ENHANCEMENTS TO THE SEI CRITERIA

The specific changes recommended for the SEI criteria are given below.

1. Change the title of the SEI evaluation criteria report. The title of the draft gives the impression that the SEI provides environment evaluations rather than a criteria for doing them.

2. Add a checklist of yes/no and short answer questions to the criteria. Using the criteria for environment evaluations is time consuming and costly, particularly if the environments are not well known to the evaluator or must be procured before the evaluations can be run. A checklist would allow the evaluator to eliminate those environments that obviously do not meet user needs. This would save time by preventing the evaluator from running complete evaluations on environments that obviously do not meet user needs. Also, a checklist would let managers reviewing evaluations quickly understand the characteristics of the environment without having to read the entire evaluation. A proposed checklist was sent to the SEI; it is included at the end of this appendix.

3. Add questions to more thoroughly evaluate the user's interface of the criteria. The user's interface is a critical area of the environment. It frequently determines whether the environment is used or sits on the shelf. Possible questions are

- a. How easy is the user's interface to learn? How easy is it to use on an on-going basis?
- b. What are the best features of the user's interface?
- c. What are the easiest-to-use features of the user's interface?
- d. What are the worst features of the user's interface?
- e. What are the hardest-to-use features of the user's interface?
- f. Does the interface have features that are likely to become frustrating as the user becomes more familiar with the tool?

4. Two other areas need additional coverage—the documentation and configuration of control reports. The TE believes these two issues have not been sufficiently covered by the criteria. Suggested questions are

- a. Is the documentation usable, well written, well formatted, and easy to understand?
- b. What reports are provided as part of the configuration control and versioning tools?

5. As much as possible, map the questions to the services contained in the NIST reference model (1991). Tying the criteria to the NIST reference model will help the criteria become more widely used. Moreover, the reference model documentation will provide additional support and clarification to the explanations given in the criteria.

PROPOSED CHECKLIST BASED ON EVALUATIONS

What hardware systems does the environment run on?

What is the minimum hardware configuration?

What operating system(s) is required?

Does it require additional software purchases such as a data base management system?

What language(s) does the environment support?

What portion(s) of the software life cycle does the environment support?

Can other vendors' tools be integrated into the environment?

What process model does the environment support?

What is the user interface methodology (X Window/Motif, X Window/Open Look, VT100-type terminals, mouse support, etc.)?

Does the environment provide a tool to prototype user interfaces for the user's applications? If not, can such a tool be integrated?

What is the purchase price (unlimited license one machine, site license)?

What is the yearly software maintenance cost?

What does the cost include?

Is training available?

What is the cost of training?

Is it easy to learn to use the environment?

Does the environment support simultaneous documentation development?

Can multiple users use the environment? Simultaneously?

Can roles be defined and assigned?

Can access to tools be controlled by user role? By user id?

What level of security does the environment provide?

Does the environment provide tools to help with configuration management? Quality assurance? Project management?

Does the environment support multiple versions of applications software?

What is needed to import and export existing code and documentation?

What tools are included as part of the environment?

What kind of documentation is provided (for example, user manual, tool integration documentation, etc.)?

What is the vendor's reputation for quality products? For customer service?

How long has the environment been offered as a commercial product?

How frequently are enhancements/new tools announced?

Are new tools a separate purchase or are they included under software maintenance?

How many organizations are presently using the environment?

Does the environment include a data base?

APPENDIX C. RISCAda EVALUATION

This appendix contains the evaluation of Telesoft's RISCAda environment, version 1. The evaluation begins with responses to the questions in the checklist proposed by NRaD (appendix B). Then, a few of the questions from sections 1 and 2 and all the questions from section 3 of the criteria are answered. The questions from sections 1 and 2 are added because the evaluator believes the responses will be helpful to people using this evaluation. The questions in sections 1 and 2 that are not included relate to an organization's readiness, product production, and project characteristics, rather than to the SEE. The same format is used in appendices D and E.

NRaD'S PROPOSED CHECKLIST

What hardware systems does the environment run on?

Sun SPARC station.

What is the minimum hardware configuration?

16 megabytes with 100 megabyte disk.

What operating system(s) is required?

SunOS 4.1 or later.

Does it require additional software purchases such as a data base management system?

X Window, version 11, release 4, must be purchased.

What language(s) does the environment support?

Ada.

What portion(s) of the software life cycle does the environment support?

Code generation.

Can other vendors' tools be integrated into the environment?

Yes, if Telesoft's ASIS (Ada Semantic Interface Specification) tool is also purchased.

What process model does the environment support?

The code generation portion of any process model can be supported.

What is the user interface methodology (X Window/Motif, X Window/Open Look, VT100-type terminals, mouse support, etc.)?

X Window/Motif.

Does the environment provide a tool to prototype user interfaces for the user's applications? If not, can such a tool be integrated?

TeleUSE (a user interface prototyping tool) is not included in the environment, but it can be purchased separately and integrated into the environment.

What is the purchase price (unlimited license, one machine, site license)?

RISCAda is priced on a per CPU basis, with prices varying by CPU type. The price for a single CPU ranges from \$6000 to \$11,500.

What is the yearly software maintenance cost?

The maintenance agreement is 20 percent of the purchase price.

What does the cost include?

The maintenance agreement provides customer support by telephone, electronic mail, or fax. It also provides the user with access to a dial-in bulletin board that contains release notes, known problems, and benchmark results.

Is training available?

Telesoft offers training on the environment and in a number of other subjects.

What is the cost of training?

The cost varies with the course taken.

Is it easy to learn to use the environment?

Yes.

Does the environment support simultaneous documentation development?

No.

Can multiple users use the environment? Simultaneously?

Yes to both questions.

Can roles be defined and assigned?

No.

Can access to tools be controlled by user role? By user id?

No, no roles are defined and users with access to the environment have access to the tools.

What level of security does the environment provide?

The normal UNIX file and directory-access-based security.

Does the environment provide tools to help with configuration management? Quality assurance? Project management?

Some baselining assistance is provided by ARCS. No quality assurance or program management tools are provided.

Does the environment support multiple versions of applications software?

Yes.

What is needed to import and export existing code and documentation?

No additional tools are required to import or export existing code.

What tools are included as part of the environment?

The purchase price includes RISCAda (compiler, library management, and associated tools); AdaTracer (a debugging tool); ARCS (design documentation and some CM tools); EZAda (a window interface to the environment); and language sensitive tools (for example, a pretty printer and a compilation order tool).

What kind of documentation is provided (for example, user manual, tool integration documentation, etc.)?

Two user's manuals are provided: one for the environment tools and one for the compiler.

What is the vendor's reputation for quality products? For customer service?

Telesoft has been producing Ada compilers since 1981 and has a good reputation for quality and customer service.

How long has the environment been offered as a commercial product?

Since August 1991.

How frequently are enhancements/new tools announced?

About every 6 months.

Are new tools a separate purchase or are they included under software maintenance?

If the tools are included in the environment, they are provided to current users as part of the software maintenance package.

How many organizations are presently using the environment?

Over 500 sites are currently using the environment.

Does the environment include a data base?

Yes, the library is based on a relational data base.

SEI CRITERIA

1.4.3 How flexible and adaptable will the software process be in the new SEE?

Since the process is defined and enforced outside the environment, using the environment does not affect its flexibility.

1.4.4 Does the environment provide process-tuning capabilities?

No, currently no tool that collects or reports process history or accounting metrics used to tune the development process is provided in the environment.

2.3.2 What degree of coupling is there between the data, control, presentation, and process integration requirements and mechanisms?

The tools are loosely coupled into the environment, allowing them to maintain individual data bases and control.

2.3.3 Can commercial off-the-shelf tools be integrated directly into the environment?

They can be integrated if the user purchases Telesoft's ASIS tool.

2.3.4 What degree of integration is required/expected between development tools and methods?

The development methods and process are enforced outside the environment, so no integration is required between the tools and methods.

2.3.5 What are the technology limitations of the integration requirements?

The integration technology allows the integration of all tools, provided the Telesoft ASIS tool is purchased.

2.4.2 What is the stability of the vendor?

Telesoft is a stable company.

2.4.3 What is the stability of the environment?

RISCAda is as stable as possible, given the immaturity of the technology field.

2.4.4 What level of integration support is expected/needed from the vendor?

The ASIS tool includes a published interface that should allow users to perform their own tool integrations. Added help is available from the Telesoft customer support line; on-site help would probably require a contract with Telesoft.

2.4.5 What is the level of product documentation quality?

It is for the most part readable and concise. The introduction, in particular, is easy to follow.

3.0 Environment Technology

3.1 Data Repository Services

3.1.1 Is the repository based on a common data storage mechanism?

Yes, the repository is the Ada library.

3.1.2 Is the repository isolated from the tools by the environment?

No, it can be accessed by any of the tools in or out of the environment.

3.1.3 Are tool changes necessary to support central data storage?

The Telesoft ASIS tool lets any tool access the library. Some work is required by the user or the tool vendor to enable the access.

3.1.4 Does the repository provide consistency checking and update services?

The repository maintains a list of Ada units that must be recompiled if the system being developed is to be in a consistent state. There is no method to check the internal state of the repository and see whether it is in a consistent state (i.e., it has not been corrupted).

3.1.5 How are changes propagated?

Changes to Ada units are automatically propagated in the repository.

3.1.6 Can the repository be distributed? Is there a data location service?

Yes, if each user on a project uses one or more separate sublibraries. The tools automatically use the version of data located in the "highest" sublibrary on the tree.

3.1.7 Is all data "live," or is there an archival service?

All data is live.

3.1.8 Can new data types be defined? New attributes?

Yes and no. The user can define new aggregate data types such as "personnel record." The user cannot define a completely new Ada unit type. This means that within an Ada function, or procedure, a new type can be declared, but the types shown in the ARCS diagrams for Ada units are set. (ARCS is an associated tool set that comes with the RISCAda environment.) For non-Ada units, such as a C procedure or library, the type can be named by the user.

3.2 Data Integration Services

3.2.1 Does the repository utilize a common data model?

Yes, the Ada library is a relational data base.

3.2.2 Can the organization query the data model? Update?

No. No.

3.2.3 Is there a registration/monitoring service for data update? Is it configurable?

No. No.

3.2.4 How are complex data queries handled?

Within RISCAda, the user can query for file names by using the same wildcard characters that UNIX provides. By using ARCS (a tool and user interface), the user can query using procedure, package, type, or variable names.

3.2.5 Are configuration services integrated with versioning services?

ARCS provides baselining and version control. However, it does not currently provide a configuration management service. These services must be imposed by management from outside the tool/environment. Plans call for future versions of ARCS to provide interface for configuration management (CM) tools.

3.2.6 Can subenvironments be defined? Distributed? Configured?

No, every user uses the same environment controls and tools, but the Ada library (data repository) can be distributed by using sublibraries so that different users have access to different data.

3.2.7 Can data be exchanged with "foreign" environments?

Source code can be imported from foreign environments and exported to foreign environments. The object code is standard object format and can be exported as such. The ARCS-created Booch diagrams are not importable or exportable in the version evaluated. According to the Telesoft representative, the newest version of ARCS lets the user incorporate the diagrams into Postscript files.

3.3 Tools

3.3.1 Which functions are under tool control, and which are under environment control?

All functions are controlled by the tools; the environment just makes access easier.

3.3.2 What expectations are made of tools?

Telesoft-provided tools can be used as they are. Other vendor tools must use the ASIS tool, which is purchased separately.

3.3.3 Can tools be integrated at varying levels?

If the user bought the ASIS and TeleUSE tools along with the RISCAda, then tools can be integrated into the library or the user interface.

3.3.4 What tools are currently integrated into the environment?

RISCAda provides the Telesoft compiler, library manager, code profiler, source dependency lister, compilation order tools, AdaTracer, and ARCS. AdaTracer is a debugging tool. ARCS is a tool box that provides a graphic system browser, a cross-referencer, a language sensitive editor, a sematisizer, a baseline manager, and a tool to automatically install source code. TeleUSE, which can be purchased separately, is a graphical user interface builder available from Telesoft.

3.3.5 Who maintains tool integrations?

Telesoft maintains its own tool integrations. Users are responsible for any tool integrations done for their site or project.

3.3.6 What techniques are used to ensure tool replaceability?

Since tools and environments are from the same vendor, it is reasonable to assume that Telesoft will ensure that current tools are replaceable by newer versions.

3.4 Task Management Services

3.4.1 Does the environment provide services for task control?

No.

3.4.2 How flexible are the control mechanisms?

Not applicable.

3.4.3 How simple is the establishment of the process model?

RISCAda and the tools it includes do not enforce any particular process model. Any particular process model would have to be overlaid on RISCAda and enforced by management without the help of RISCAda. On the other hand, RISCAda does not prevent the use of a process model.

If an organization's process model requires the use of specific in-house tools, it will have to be used outside the RISCAda environment. It cannot be integrated into the environment.

3.4.4 Can a subenvironment be specified for individual tools?

No, RISCAda cannot specify subenvironments.

3.4.5 Can roles be defined for individual users?

No, although the user in ARCS can be assigned access based on whether he or she is a baseline coordinator or a normal ARCS user. The baseline coordinator has greater access than a normal ARCS user.

3.5 Message Services

3.5.1 What type of message service is provided?

No message service is currently provided. Plans call for the support of SoftBench and ToolTalk in the future.

3.5.2 Are message services handled as "transactions"? Can messages be canceled? Re-sent?

Not applicable.

3.5.3 Can a "process flow" be defined?

No, not within the environment.

3.5.4 Can message services span environment instances?

Not applicable.

3.5.5 Can third-party tools be integrated into the system?

Yes, if the user bought additional tools ASIS allows integration into the library; TeleUSE allows integration into the user interface.

3.6 User Interface Services

3.6.1 Is the windowing system in keeping with industry trends?

Yes, the windowing system is based on X Window/Motif.

3.6.2 What services are offered?

Normal X Window and Motif services are offered.

3.6.3 How are external tools integrated into the interface?

Telesoft provides the integration "hooks" for Telesoft tools such as TRIAD, a tool for developing realtime systems in Ada. Non-Telesoft tools can be integrated by using ASIS and TeleUSE. ASIS and TeleUSE must be purchased.

3.6.4 How flexible is the user interface?

The interface for the main functions is set. However, users can define icons to retain frequently used command qualifier sets. The ARCS menus can be customized.

3.7 Security

3.7.1 What types of security are provided?

The UNIX security primitives are provided. ARCS baselining functions provide additional security, including the use of check-in, check-out procedures for multi-user access to program libraries.

3.7.2 Are reports provided?

No.

3.7.3 At what level of granularity is security provided?

Security is provided at the library, library element, and file levels.

3.8 Framework Administration and Configuration

3.8.1 What personnel are required for administration?

RISCAda requires, at a minimum, someone to load it, communicate with Telesoft personnel, and be the point-of-contact for users' questions. If it is used in a network configuration, someone knowledgeable about the network is also required (though these functions could be accomplished by the same person). In addition, if the ARCS baselining capability is used, a baseline coordinator is needed.

3.8.2 Is training provided?

Yes, Telesoft provides training in the use of its tools and offers additional training classes in Ada and object-oriented programming.

3.8.3 How easily can the environment be installed? Rebuilt? Upgraded?

Load the installation tape, which is in tar format, and define some system variables in order to install the environment on Sun systems. After that, individual users must change their login files. Rebuilds and upgrades should require no more effort. The only exception to this occurs when the upgrade includes changes to the library definition, in which case the users' old libraries would also have to be updated to the new format. Telesoft would presumably provide a conversion routine to handle this update.

3.8.4 Is the environment stable?

Yes.

3.8.5 What enhancements are envisioned?

Telesoft plans to introduce a cross compiler and TRIADS (to provide support for realtime systems development). Also, plans call for the integration of CM tools and

front-end computer-aided software engineering (CASE) tools. Telesoft has recently signed an agreement with CASEWARE that will allow Telesoft to integrate CASEWARE CM into RISCAda, thus providing configuration management capability.

3.8.6 Are utilities provided for conversion from other tools and environments?

Source code files created with other tools and environments would have to be compiled into the library in order to be accessed by the RISCAda tools.

3.9 Process Encoding

3.9.1 What mechanisms are used for process encoding?

The process is encoded by data sharing between tools and tool activation, but the order of activation and the process itself have to be enforced by organization/project management. RISCAda cannot automatically enforce itself from within the environment.

3.9.2 Is the environment oriented toward any one process? Portion of the life cycle?

The environment is oriented toward the code development, testing, and maintenance portions of the life cycle. The design can be documented by using ARCS, but the design development process is not supported.

3.9.3 How flexible is process support?

It is extremely flexible; RISCAda can support any process that the using organization wants it to. It cannot be used to enforce the process.

3.9.4 What level of effort is required to encode an organization's process?

An organization's process is not encoded into the environment; it is enforced by management from outside. It will be possible to encode a portion of the process once CM tools have been integrated into RISCAda.

3.9.5 How complete is the process encoding?

See the answer to 3.9.4.

APPENDIX D. RATIONAL EVALUATION

The evaluation was conducted on version 12.4.2 of the Rational environment.

NRaD'S PROPOSED CHECKLIST

What hardware systems does the environment run on?

The Rational environment runs on a Rational machine.

What is the minimum hardware configuration?

The minimum hardware configuration requirement is one Rational machine.

What operating system(s) is required?

The operating system for the Rational machine is proprietary to the company.

Does it require additional software purchases such as a data base management system?

No.

What language(s) does the environment support?

Ada.

What portion(s) of the software life cycle does the environment support?

The environment gives some support to the entire life cycle, but it gives the strongest support to code generation and unit testing.

Can other vendors' tools be integrated into the environment?

Yes, by using the Language Reference Manual (LRM) interfaces that are included in the environment.

What process model does the environment support?

As delivered, the environment supports the development process specified in DoD-Std-2167A (Military Standard Defense System Software Development, February 1988), but the user can modify the environment to support any process.

What is the user interface methodology (X Window/Motif, X Window/Open Look, VT100-type terminals, mouse support, etc.)?

Currently, the user interface methodology supports VT100-type terminals; plans call for an X Window interface in September 1992.

Does the environment provide a tool to prototype user interfaces for the user's applications? If not, can such a tool be integrated?

No, a user interface prototyping tool is not included, but one can be integrated.

What is the purchase price (unlimited license, one machine, site license)?

Rational does not sell site licenses. One machine costs \$48,500. To allow an unlimited number of users, the machine would cost an additional \$300,000.

What is the yearly software maintenance cost?

\$42,000.

What does the cost include?

The maintenance cost provides installation services, a telephone and electronic mail contact for questions, local and remote diagnostics services, and preventive and corrective maintenance of the environment and machine.

Is training available?

Yes, a number of training courses are provided.

What is the cost of training?

The cost of training varies depending on the course. The Fundamentals of the Rational Environment course costs \$8750 for 12 students.

Is it easy to learn to use the environment?

No, the training class is required to learn the environment.

Does the environment support simultaneous documentation development?

Yes.

Can multiple users use the environment? Simultaneously?

Yes, if additional sessions (licenses) are purchased, then multiple users can use the environment simultaneously.

Can roles be defined and assigned?

No.

Can access to tools be controlled by user role? By user id?

Access cannot be controlled by user role, but access can be controlled by user id.

What level of security does the environment provide?

The environment allows or denies access to tools, files, or Ada units on the basis of user ids. Access to the environment is password protected.

Does the environment provide tools to help with configuration management? Quality assurance? Project management?

The environment provides configuration management and program management tools. No quality assurance tools are provided with the environment, but they can be integrated later.

Does the environment support multiple versions of applications software?

Yes.

What is needed to import and export existing code and documentation?

The environment provides all tools required to import and export code and documentation.

What tools are included as part of the environment?

The Rational environment includes an Ada compiler; a debugger; a configuration management and version control tool; a library manager; a language sensitive editor (including a pretty printer); some document generation capability; a report generation tool (for generating configuration management and project management reports); and a tool integration support tool.

What kind of documentation is provided (for example, user manual, tool integration documentation, etc.)?

An extensive set of user and reference manuals is provided.

What is the vendor's reputation for quality products? For customer service?

Rational has been in business since 1980 and has a good reputation for both quality and customer service.

How long has the environment been offered as a commercial product?

The environment has been a commercial product since 1985.

How frequently are enhancements/new tools announced?

About every 6 months.

Are new tools a separate purchase or are they included under software maintenance?

If new tools are made part of the environment, they are received under the maintenance agreement, otherwise they are a separate purchase.

How many organizations are presently using the environment?

Over 300 sites are currently using the environment.

Does the environment include a data base?

Yes.

SEI CRITERIA

1.4.3 How flexible and adaptable will the software process be in the new SEE?

The environment manager can modify or override the defined process rather easily.

1.4.4 Does the environment provide process-tuning capabilities?

No, however, third-party tools that let users collect process history and accounting metrics tune the process can be integrated.

2.3.2 What degree of coupling is there between the data, control, presentation, and process integration requirements and mechanisms?

The environment controls tools integrated into it. It requires, for example, that tools interface with the central data repository.

2.3.3 Can commercial off-the-shelf tools be integrated directly into the environment?

Yes, by using the Rational interface tool that is provided.

2.3.4 What degree of integration is required/expected between development tools and methods?

The development process methods are defined in and controlled by the environment.

2.3.5 What are the technology limitations of the integration requirements?

The integration requirements let third-party tools be integrated into Rational without requiring further cooperation from the tool developers.

2.4.2 What is the stability of the vendor?

Rational is a stable company.

2.4.3 What is the stability of the environment?

The Rational environment is as stable as possible, given the immaturity of the technology field.

2.4.4 What level of integration support is expected/needed from the vendor?

By using the Rational interface, the user should be able to integrate most tools into the environment. If required, help can be contracted from Rational.

2.4.5 What is the level of product documentation quality?

The product documentation is extensive and comprehensive. In a few cases, the index lists commands only under their Rational procedure name, which can make them difficult to find if the user does not know the name of the procedure.

3.0 Environment Technology

3.1 Data Repository Services

3.1.1 Is the repository based on a common data storage mechanism?

Yes, it is based on a Distributed Intermediate Attributed Notation for Ada (DIANA) data base.

3.1.2 Is the repository isolated from the tools by the environment?

Yes.

3.1.3 Are tool changes necessary to support central data storage?

Not necessarily, current tools can be integrated into the repository by using the Language Reference Manual (LRM) interfaces provided by Rational (American National Standards Institute, Inc., 1990).

3.1.4 Does the repository provide consistency checking and update services?

If a user has reason to suspect the repository is inconsistent, the operator can run a procedure that checks consistency. The results can be used to restore consistency. However, according to the vendor, it is unusual for the data base to become inconsistent unless there is a hardware failure.

3.1.5 How are changes propagated?

The user is automatically notified of changes, then he or she must ensure that the changes are propagated to other entities.

3.1.6 Can the repository be distributed? Is there a data location service?

The repository resides on a Rational machine. It can be distributed across a number of Rational machines if required. There is a data location service, provided the data location is in the search list of the user's home library.

3.1.7 Is all data "live," or is there an archival service?

There is an archival service.

3.1.8 Can new data types be defined? New attributes?

Within Ada units, new data types can be defined under the normal rules of Ada. The data types used by the repository are Ada units, libraries of various kinds, and text files. The user can use those types to define new types for the repository by using the Ada rules for defining new data types.

3.2 Data Integration Services

3.2.1 Does the repository utilize a common data model?

All data in the repository are organized in DIANA trees.

3.2.2 Can the organization query the data model? Update?

The data model can be queried only by using a specific tool, the DIANA object manager. It is not possible for the user to directly modify the data base; the tools must be used.

3.2.3 Is there a registration/monitoring service for data update? Is it configurable?

The data base keeps track of the state of data base objects. When there is a change of state, for example, from source to compiled, the environment automatically carries out a number of actions. The user can define new or additional actions for state changes or data updates.

3.2.4 How are complex data queries handled?

Most data queries can be handled in a single pass. Some extremely complex ones may require extra passes through the data base.

3.2.5 Are configuration services integrated with versioning services?

Yes, a configuration management and version control system comes with the environment.

3.2.6 Can subenvironments be defined? Distributed? Configured?

Yes, subenvironments can be created whose access is controlled by project or user information. The subenvironments can be distributed over a number of interconnected Rational machines.

The subenvironments can be configured. The parameters for some tools can be reconfigured for each subenvironment. Access to tools can vary. New tools can be integrated into subenvironments as required.

3.2.7 Can data be exchanged with "foreign" environments?

Text files and Ada units can be easily imported and exported. If a cross-compiler has been integrated into the environment, object code generated by it can be exported to an environment with the appropriate compiler.

3.3 Tools

3.3.1 Which functions are under tool control, and which are under environment control?

The environment controls access to the data repository. The tools carry out their primary function. However, some tools interact with the data repository throughout their operation. For example, the compiler interacts with the data repository throughout the process of compiling, rather than waiting until completion to update the repository.

3.3.2 What expectations are made of tools?

They are expected to provide their resulting data in a format that can be stored in the data repository.

3.3.3 Can tools be integrated at varying levels?

Rational integrates the tools it develops into the data repository. Non-Rational tools can be integrated into the data repository by using the LRM interfaces provided by Rational (American National Standards Institute, Inc., 1990). They can also be integrated at the user interface level if desired, although this high-level integration is probably less useful. When tools are integrated only at the user interface level, the tools do not interact directly with the environment; the user must initiate all interactions.

3.3.4 What tools are currently integrated into the environment?

The Rational environment contains a compiler, debugger, pretty printer, cross referencer, data base manager, language sensitive editor, and configuration control tool. The user can also purchase Rational design tools. Several of the existing third-party compilers and cross compilers have been integrated into the Rational environment. Additionally, Cadre Teamwork, Interleaf, AdaMAT/R, and an X Window/Motif Interface tool from Screen Machine have been integrated.

3.3.5 Who maintains tool integrations?

Integrations done by Rational are maintained by Rational. Those done by the user are maintained by the user, although Rational's support services would probably assist with problems.

3.3.6 What techniques are used to ensure tool replaceability?

Tool replaceability is not a major problem because tools access the repository and the environment through specific, defined interfaces. If a tool must be replaced by either a completely new tool or a new version of the old tool, the user has to use the LRM interfaces to integrate the new tool (American National Standards Institute, Inc., 1990). It would be essentially the same task as the original integration.

3.4 Task Management Services

3.4.1 Does the environment provide services for task control?

Yes.

3.4.2 How flexible are the control mechanisms?

The environment manager can write scripts to create a collection of operations that trigger tools in a specified order after completion of a user action.

3.4.3 How simple is the establishment of the process model?

Generate a script to establish the process model (see 3.4.2 above), then the environment will enforce the process as given.

3.4.4 Can a subenvironment be specified for individual tools?

No, only the one environment exists. It is not possible to create a subenvironment to suit the needs of a specific tool. The Rational environment does allow a user to define a subenvironment and integrate certain tools into it. The subenvironment can have a unique task-ordering method. This technique accomplishes almost the same thing as defining a subenvironment for an individual tool.

3.4.5 Can roles be defined for individual users?

Yes and no. It is possible to grant or deny access to tools and data by user id. But it is not possible to assign a specific role to the user id and have access assigned by that action.

3.5 Message Services

3.5.1 What type of message service is provided?

Users can send electronic mail messages to each other or to groups. The tools cannot send messages to each other beyond error interrupts. The tools communicate primarily through the data repository.

3.5.2 Are message services handled as "transactions"? Can messages be canceled? Re-sent?

Error interrupts are broadcast as soon as the error is found/occurs, therefore it cannot be canceled. Re-sending that message is also impossible, although the user can cause an identical message to be generated by calling the tool again without fixing the error.

User-to-user messages are treated just like other electronic mail messages.

3.5.3 Can a "process flow" be defined?

To an extent, the process flow is already defined. Code has to go through a defined process before it can be unit tested. The Rational does allow the user to customize the process flow by using work orders and the design facility. This lets the project manager define a sequence of steps that must be executed in the event of a given action.

3.5.4 Can message services span environment instances?

If the other environment instance is another project's environment on the same machine or the environment of another networked Rational machine, then the messages can be sent to other environments. Otherwise, the messages can not be sent to other environment instances.

3.5.5 Can third-party tools be integrated into the system?

Yes, third-party tools can be integrated into the environment by using the interfaces provided. These interfaces let the tools interface with the repository.

3.6 User Interface Services

3.6.1 Is the windowing system in keeping with industry trends?

No, it is based on text character windows and the use of key combinations for actions.

NOTE: Just before this report was issued, the TE learned that Rational was delivered a new Motif-based user interface in September 1992. This new version of Rational was not available for evaluation in time to be included in this report. All responses concerning the user interface are based on version 12.4.2.

3.6.2 What services are offered?

The services required for the use of the environment and tools are provided. There are no services provided to aid the integration of new tools.

3.6.3 How are external tools integrated into the interface?

The integration is done using the LRM interfaces provided by Rational (American National Standards Institute, Inc., 1990).

3.6.4 How flexible is the user interface?

The user can assign actions to key stroke sequences as desired. That is all the flexibility provided by the user interface.

3.7 Security

3.7.1 What types of security are provided?

Individual files and/or directories (libraries) can be given access protection to prevent unauthorized access. In addition, access to tools can be controlled through the use of commands provided by the environment.

3.7.2 Are reports provided?

A user with access can generate reports on who accessed what unit. If the person performing the access provides the data, the report can also include what action was performed on the unit. If the project uses the work order method provided by Rational, the system automatically tracks who changed what, what the change was, etc., and provides reports on command.

3.7.3 At what level of granularity is security provided?

Access to information in the data base can be controlled at several levels: the entire data base, a specific subsystem or library, or a specific unit (procedure, package, function).

3.8 Framework Administration and Configuration

3.8.1 What personnel are required for administration?

Someone is required to administer and maintain the system, i.e., to add new users, delete old users, etc. A maintenance agreement with the vendor is required for more than routine upkeep of the environment.

3.8.2 Is training provided?

Yes, for an additional cost, classes are provided on how to use the system.

3.8.3 How easily can the environment be installed? Rebuilt? Upgraded?

The environment needs to be installed by Rational personnel. Rebuilds and upgrades will also require help from Rational personnel.

3.8.4 Is the environment stable?

Yes. Changes are envisioned over time, but the current configuration will continue to be supported.

3.8.5 What enhancements are envisioned?

In September 1992, a new version of the environment called Rational_Access became available. It provides a Motif interface to the Rational. In addition, plans exist to move from the vendor-specific hardware requirement.

3.8.6 Are utilities provided for conversion from other tools and environments?

Tools exist to help import source code and text files from other environments.

3.9 Process Encoding

3.9.1 What mechanisms are used for process encoding?

The mechanisms are data sharing between tools and support for tool activation. Data sharing is the primary method of process encoding.

3.9.2 Is the environment oriented toward any one process? Portion of the life cycle?

The environment is tilted slightly to object-oriented development, although other development methods are supported. Also, development under MIL-STD-2167A is fully supported.

The environment supports activities under all portions of the life cycle. The coverage of the code and unit test portion is the most complete. The support can be expanded by integrating third-party tools or buying additional Rational tools.

3.9.3 How flexible is process support?

The process support is flexible enough to allow changes to all defineable aspects. Coding standards, design, document production, and configuration control can be modified through the interfaces provided.

3.9.4 What level of effort is required to encode an organization's process?

Encoding an organization's process is easy if it just requires changing the order of tool activation or the tools a user can access. Also, the project can enforce coding standards, design, and change control requirements.

3.9.5 How complete is the process encoding?

The encoding of the source code development process is fairly complete, although some portions of an organization's process may have to be enforced by management rather than the environment.

APPENDIX E. SUN ADA DEVELOPMENT ENVIRONMENT EVALUATION

The evaluation was conducted on version 1.2 of the environment.

NRaD'S PROPOSED CHECKLIST

What hardware systems does the environment run on?

A Sun 4 or SPARCStation 1 workstation.

What is the minimum hardware configuration?

12 megabytes of physical memory, 30 megabytes of swap space, and 33 megabytes of disk space.

What operating system(s) is required?

SunOS 4.1 or later.

Does it require additional software purchases such as a data base management system?

Open Windows version 2FCS and an X11/NeWs server.

What language(s) does the environment support?

Ada.

What portion(s) of the software life cycle does the environment support?

Code generation.

Can other vendors' tools be integrated into the environment?

No.

What process model does the environment support?

The environment can support the code generation portion of any process model.

What is the user interface methodology (X Window/Motif, X Window/Open Look, VT100-type terminals, mouse support, etc.)?

X Window/Open Look.

Does the environment provide a tool to prototype user interfaces for the user's applications? If not, can such a tool be integrated?

No, no such tool is provided and tools cannot be integrated into the Sun ADE.

What is the purchase price (unlimited license one machine, site license)?

A single floating license costs \$10,000; a 10-user floating license costs \$80,000.

What is the yearly software maintenance cost?

For the single license, maintenance is \$160; for the 10-user license, maintenance is \$1120.

What does the cost include?

Maintenance includes all environment upgrades and phone-in technical support with an 800 number.

Is training available?

Yes, training is available on a variety of subjects including the environment.

What is the cost of training?

Environment training costs approximately \$1700.

Is it easy to learn to use the environment?

Yes.

Does the environment support simultaneous documentation development?

No.

Can multiple users use the environment? Simultaneously?

Yes, multiple users can use the environment.

Can roles be defined and assigned?

No.

Can access to tools be controlled by user role? By user id?

No, users who can access the environment can access all tools in the environment.

What level of security does the environment provide?

Security is provided by the UNIX security primitives.

Does the environment provide tools to help with configuration management? Quality assurance? Project management?

No, the environment provides no configuration management, quality assurance, or project management tools.

Does the environment support multiple versions of applications software?

No.

What is needed to import and export existing code and documentation?

A command is provided to import code into the environment.

What tools are included as part of the environment?

The environment includes the Verdix compiler; a debugger; editing tools, including a pretty printer; library management tools; and tools that determine the dependencies of the Ada units.

What kind of documentation is provided (for example, user manual, tool integration documentation, etc.)?

User manuals are provided for both the environment and the compiler.

What is the vendor's reputation for quality products? For customer service?

Both Verdix and Sun have good reputations for quality and service.

How long has the environment been offered as a commercial product?

The environment has been a commercial product since 1990.

How frequently are enhancements/new tools announced?

That depends on the Verdix compiler.

Are new tools a separate purchase or are they included under software maintenance?

If the tools were sold to new users as a part of the environment, they are included in the upgrades as part of software maintenance.

How many organizations are presently using the environment?

Approximately 500 organizations are using the environment.

Does the environment include a data base?

No.

SEI CRITERIA

1.4.3 How flexible and adaptable will the software process be in the new SEE?

Since the process is defined and enforced outside the environment, the environment does not affect the flexibility of the process.

1.4.4 Does the environment provide process-tuning capabilities?

No, currently no tool that collects or reports process history or accounting metrics for the purpose of tuning the development process is provided in the environment.

2.3.2 What degree of coupling is there between the data, control, presentation, and process integration requirements and mechanisms?

The tools are loosely coupled with the environment.

2.3.3 Can commercial off-the-shelf tools be integrated directly into the environment?

Yes, the interface-package-provided COTS tools can interface with the environment repository.

2.3.4 What degree of integration is required/expected between development tools and methods?

The development process methods are enforced outside the environment, so no integration is required between the tools and methods.

2.3.5 What are the technology limitations of the integration requirements?

Tools can be integrated to a level that allows interface with the DIANA intermediate language. Verdix is developing an ASIS interface to allow deeper integration.

2.4.2 What is the stability of the vendor?

Verdix and Sun are both stable companies.

2.4.3 What is the stability of the environment?

The Sun Ada Development Environment (Sun ADE) is as stable as possible, given the immaturity of the technology field.

2.4.4 What level of integration support is expected/needed from the vendor?

The user can integrate on-site tools by using the interface package provided. Any questions can be answered by Verdix Customer Support personnel. On-site support will require a contract with Verdix.

2.4.5 What is the level of product documentation quality?

For the most part, the documentation is well organized and understandable, although a novice environment user may have difficulty finding required information. That is particularly true for the introduction and tutorials. The command reference sections are formatted similarly to the main pages on the UNIX operating system (OS).

3.0 Environment Technology

3.1 Data Repository Services

3.1.1 Is the repository based on a common data storage mechanism?

The data repository is the Ada library. It stores the object code for the software product. The Ada library is based on the UNIX file structure.

3.1.2 Is the repository isolated from the tools by the environment?

No, the tools can access the repository either through the environment or outside of it.

3.1.3 Are tool changes necessary to support central data storage?

The user will be required to acquire tools that can access the Sun Ada library. Since the Sun Ada Development Environment (Sun ADE) is a front-end to the Verdix Ada compiler, Verdix tools should be usable throughout the environment.

3.1.4 Does the repository provide consistency checking and update services?

The repository verifies that new code is consistent with old. It does not provide tools that allow the user to check the consistency of the repository.

3.1.5 How are changes propagated?

Changes to the data repository caused by updates of the environment are propagated by using tools/software provided by the vendor during update installation.

3.1.6 Can the repository be distributed? Is there a data location service?

Each user can have his or her own library and the environment can be used across a network.

3.1.7 Is all data "live," or is there an archival service?

Files can be archived by using the method provided by the host system. Then the data can be used in an archived library within the environment.

3.1.8 Can new data types be defined? New attributes?

New Ada types can be defined within the Ada units. The data types used by the repository, Ada units, object files, etc., are defined and cannot be redefined or augmented.

3.2 Data Integration Services

3.2.1 Does the repository utilize a common data model?

The Ada library is based on the DIANA data model.

3.2.2 Can the organization query the data model? Update?

No. No. The user can use a package provided in the Sun ADE to let tools interface with the data model.

3.2.3 Is there a registration/monitoring service for data update? Is it configurable?

No. No.

3.2.4 How are complex data queries handled?

The user can search by unit types, unit names, or a combination of the two.

3.2.5 Are configuration services integrated with versioning services?

No, configuration control and versioning services are supplied by the Sun ADE.

3.2.6 Can subenvironments be defined? Distributed? Configured?

No. It is possible for a project/environment manager to control a user's access to particular files or an Ada unit, but there is no way to prevent a user from using a tool. Nor is there any way to define a subenvironment to allow or prevent such access. The environment can be distributed to the extent that it can be used across the network.

3.2.7 Can data be exchanged with "foreign" environments?

Yes, there is a method for importing and exporting source code files. There is no method to import or export object files.

3.3 Tools

3.3.1 Which functions are under tool control, and which are under environment control?

All functions are under tool control. The environment provides a simpler way to use the tools.

3.3.2 What expectations are made of tools?

Tools are expected to be able to access the Ada library and produce output in the correct format to be stored in the Ada library.

3.3.3 Can tools be integrated at varying levels?

The user can use a provided interface package to integrate tools into the environment at a level that allows the tools to interface with the repository.

3.3.4 What tools are currently integrated into the environment?

The environment comes with code development tools, including the Verdex compiler, linker, pretty printer, debugger, editor, and library manager. Additionally, an on-line LRM, graphical software architecture viewer, and task browser are provided.

3.3.5 Who maintains tool integrations?

Tools are integrated by Sun and the integrations are maintained by Sun.

3.3.6 What techniques are used to ensure tool replaceability?

Since the integrated tools are provided and integrated by Sun, Sun ensures tool replaceability.

3.4 Task Management Services

3.4.1 Does the environment provide services for task control?

No.

3.4.2 How flexible are the control mechanisms?

Not applicable.

3.4.3 How simple is the establishment of the process model?

Since the environment does not provide task control services, the user's process model will have to be encoded through management control external to the environment.

3.4.4 Can a subenvironment be specified for individual tools?

No.

3.4.5 Can roles be defined for individual users?

No.

3.5 Message Services

3.5.1 What type of message service is provided?

User-to-user messages are handled by the development platform's standard mail system. Tool-to-tool messages are not handled. The user can invoke tools using the drag and drop features of an Open Look environment.

3.5.2 Are message services handled as "transactions"? Can messages be canceled? Re-sent?

Not applicable.

3.5.3 Can a "process flow" be defined?

A process flow cannot be defined within the environment; the process flow would have to be defined and enforced external to the environment.

3.5.4 Can message services span environment instances?

This question is not applicable because it is not possible to define environment instances.

3.5.5 Can third-party tools be integrated into the system?

No.

3.6 User Interface Services

3.6.1 Is the windowing system in keeping with industry trends?

Yes, the windowing system is based on the Open Look Graphical User Interface (GUI) specification.

3.6.2 What services are offered?

All Open Look services are provided.

3.6.3 How are external tools integrated into the interface?

External tools are not integrated into the interface or the environment.

3.6.4 How flexible is the user interface?

The user can modify the user interface as allowed by Open Look (i.e., colors, sizing, etc.).

3.7 Security

3.7.1 What types of security are provided?

Security is provided by the UNIX operating system. No additional security is provided by the environment.

3.7.2 Are reports provided?

No.

3.7.3 At what level of granularity is security provided?

Security is provided at the file and directory levels.

3.8 Framework Administration and Configuration

3.8.1 What personnel are required for administration?

An environment manager is required to install and maintain the system, to provide user access, and answer questions. This does not have to be a full-time position and can probably be done by the system administrator.

3.8.2 Is training provided?

Verdix, the developer of the tools in the environment, provides a number of training courses ranging from Ada to Software Engineering, as well as courses on

specific tools. Ada training is also provided by Sun. It is the opinion of the evaluator that no training on the Sun ADE is needed because it is easy to learn. A tutorial is included in the user's manual.

3.8.3 How easily can the environment be installed? Rebuilt? Upgraded?

Installation requires reading a tape into the system and setting some system variables. Under normal circumstances, installation is not difficult. Rebuilding and/or upgrading the system is a similar process.

3.8.4 Is the environment stable?

Yes, given the relative lack of maturity of the technology.

3.8.5 What enhancements are envisioned?

A capability to allow the user to integrate new tools into the user interface will be offered in the future.

3.8.6 Are utilities provided for conversion from other tools and environments?

An import and export function is provided to be used with the source code files.

3.9 Process Encoding

3.9.1 What mechanisms are used for process encoding?

The mechanism provided by the environment is data sharing among tools. The environment provides minimal enforcement of the process encoding. Enforcement is limited to that required for the tools to work properly; for example, an Ada unit must be compiled before it can be linked.

3.9.2 Is the environment oriented toward any one process? Portion of the life cycle?

The environment supports only the code development portion of the life cycle. The process used to design or develop the code is left to the user. The environment supports the waterfall model, spiral model, and others equally well.

3.9.3 How flexible is process support?

Since the process enforcement is minimal, the process support is very flexible.

3.9.4 What level of effort is required to encode an organization's process?

The environment cannot be modified to encode an organization's software development process. The process will have to be enforced from outside of the environment by project management.

3.9.5 How complete is the process encoding?

The process encoding of a standard code development process is reasonably complete, with the exception that no configuration management or versioning capabilities are provided. Additionally, the project management has no way of enforcing a requirement for the simultaneous project documentation and code development.

APPENDIX F. REFERENCES

- American National Standards Institute, Inc. 1990. "Appendix F for the R1000 Target." In *Reference Manual for the Ada Programming Language*, ANSI/MIL-STD-1815A-1983. Rational Corporation, Santa Clara, CA.
- NIST. 1991. "NIST Reference Model for Frameworks of Software Engineering Environments," NIST Special Publication 500-201, Technical Report ECMA TR55, 2nd ed. NIST, Gaithersburg, MD.
- Smith, D., C. Huff, E. Morris, and P. Zarrella. 1991. "Evaluation of Software Engineering Environments," Draft (December). Software Engineering Institute, Pittsburgh, PA.

REPORT DOCUMENTATION PAGEForm Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE January 1993	3. REPORT TYPE AND DATES COVERED Final: FY 1992
4. TITLE AND SUBTITLE EVALUATIONS USING THE SOFTWARE ENGINEERING INSTITUTE (SEI) ENVIRONMENTS CRITERIA		5. FUNDING NUMBERS PE: 0602234N	
6. AUTHOR(S) S. A. Parker			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Command, Control and Ocean Surveillance Center (NCCOSC) RDT&E Division (NRaD) San Diego, CA 92152-5000		8. PERFORMING ORGANIZATION REPORT NUMBER NRaD TD 2448	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Office of Naval Technology Arlington, VA 22217-5000		10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES			
12a. DISTRIBUTION/AVAILABILITY STATEMENT Authorized for public release; distribution is unlimited.		12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) This investigation, performed by NRaD under the Software Engineering Environment Prototypes (SWEEP) task of the software engineering for the Command, Control, and Communications (C ³) Systems project, analyzed strengths and weaknesses in the Software Engineering Institute's (SEI) criteria for evaluating software engineering environments (SEEs). The SEI's criteria were used to evaluate three commercial software engineering environments: the Rational, RISCAda, and Sun Ada Development Environment. The evaluations of the engineering environments discovered minor weaknesses in the SEI criteria; the SEI will modify its SEE evaluation criteria according to NRaD's recommendations before publishing the final draft of the criteria.			
14. SUBJECT TERMS Ada RISCAda Sun Ada Development Environment Rational environment			15. NUMBER OF PAGES 71
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT SAME AS REPORT

UNCLASSIFIED

21a. NAME OF RESPONSIBLE INDIVIDUAL S. A. Parker	21b. TELEPHONE (include Area Code) (619) 553-5120	21c. OFFICE SYMBOL Code 411

INITIAL DISTRIBUTION

Code 01	R. T. Shearer	(1)
Code 0012	Patent Counsel	(1)
Code 0142	K. J. Campbell	(1)
Code 144	V. Ware	(1)
Code 40	R. C. Kolb	(1)
Code 402	R. A. Wasilausky	(1)
Code 41	A. G. Justice	(1)
Code 411	J. Schulte	(1)
Code 411	H. Mumm	(1)
Code 411	S. Parker	(10)
Code 834	B. Barlin	(1)
Code 961	Archive/Stock	(6)
Code 964B	Library	(2)

OUSD(A) Arlington, VA 20301-3080		Naval Surface Warfare Center Silver Spring, MD 20903-5000	(3)
Defense Advanced Research Projects Agency Arlington, VA 22203-1714	(2)	Washington Navy Yard Washington, DC 20375-5000	(2)
Defense Technical Information Center Alexandria, VA 22304-6145	(4)	Center for Naval Analyses Alexandria, VA 22302-0268	
Chief of Naval Operations Washington, DC 20350-2000		GIDEP Operations Center Corona, CA 91718-8000	
Office of the Chief of Naval Research Arlington, VA 22217-5000	(2)	NCCOSC Washington Liaison Office Washington, DC 20363-5100	
Office of Naval Technology Arlington, VA 22217-5000	(3)	Navy Acquisition, Research & Development Information Center (NARDIC) Washington, DC 20360-5000	
Naval Research Laboratory Washington, DC 20375-5000	(4)	ADA Joint Program Office Washington, DC 20301-3081	
Naval Undersea Warfare Center Newport, RI 02841-5047	(3)	U. S. Army, Headquarters Washington, DC 20301-5000	
Space and Naval Warfare Systems Command Washington, DC 20363-5100	(3)	Rome Laboratory Griffiss AFB, NY 13441-5700	
Naval Command, Control and Ocean Surveillance Center San Diego, CA 92147-5081		STARS Technology Center Arlington, VA 22203	
Naval Air Warfare Center Weapons Division China Lake, CA 93555-6001		U. S. Army Ballistic Research Laboratory Aberdeen, MD 21005-5066	
		Georgia Institute of Technology Atlanta, GA 30332	